IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

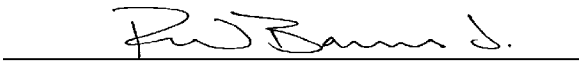
Serial Number:	09/852,200	Applicant:	Beckwith et al.
Filed:	2001.05.08	Examiner:	Michael D. Meucci
Group Art Unit:	2142	Customer Number:	**24,319**
Attorney Docket:	B1-4171	Title:	Multi-Client to Multi-Server Simulation Environment Control System (JULEP)

APPLICANTS' BRIEF ON APPEAL

Sir:

Pursuant to 37 C.F.R. 41.37, applicants hereby submit this brief on appeal in response to the examiner's rejection of claims 7-8, 10-17, 20, 22-23, 27, and 41 in the final office action of 2006.03.06, and the Notice of Panel Decision from Pre-Appeal Brief Review dated 2006.07.19. The Commissioner is authorized to charge any fees associated with the filing of this brief, including extensions of time, to the LSI Logic Corporation deposit account number 12-2252.

2006.08.02
Date


Rick Barnes, 39,596

59724us-topto-20060801-AppealBrief.doc

I. REAL PARTY IN INTEREST

The real party in interest is LSI Logic Corporation, a corporation of Delaware, and assignee of record of the entire right, title, and interest in and to the invention and application for patent thereon from the inventors Robert Beckwith, Robert Sanzone, and Mary Albanese.

II. RELATED APPEALS AND INTERFERENCES

There are no related appeals or interferences.

III. STATUS OF CLAIMS

Claims 7-8, 10-17, 20, 22-23, 27, 30-33, 36, 38, and 41-46 are in the case. Claims 7-8, 10-17, 20, 22-23, 27, and 41 are rejected under 35 USC § 103. Claims 30-33, 36, 38, and 42-46 have been withdrawn from consideration. The claims on appeal are claims 7-8, 10-17, 20, 22-23, 27, and 41, as given in the Appendix.

IV. STATUS OF AMENDMENTS

No amendments after final rejection have been filed.

V. SUMMARY OF CLAIMED SUBJECT MATTER

GROUPING OF CLAIMS

Appellants consider that the claims on appeal do not stand or fall together in regard to the rejections made over the cited references. Accordingly, and for purposes of this appeal only, appellants suggest that the patentable subject matter of the application falls into six groups as given below. This subsection is included at this point so that a summary of claimed subject matter for the independent claim and a representative dependent claim of each of the different groups that is argued separately can be provided hereafter.

Group I consisting of claims 41, 10, 20, and 22-23, directed to the ***broader embodiments*** of the present invention, reciting, *inter alia*, at least one server process, at

least one client process, and only one control process, which are all separate and distinct processes.

Group II consisting of claims 7 and 11, additionally directed to embodiments in which ***ordered queues*** are implemented, and having a separate patentability as described more completely hereafter.

Group III consisting of claims 8, 16, and 27, additionally directed to embodiments in which ***event expressions*** are implemented, and having a separate patentability as described more completely hereafter.

Group IV consisting of claims 12-13 and 17, additionally directed to embodiments in which ***both ordered queues and event expressions*** are implemented, and having a separate patentability as described more completely hereafter.

Group V consisting of claims 14-15, additionally directed to embodiments in which ***finish messages*** are implemented, and having a separate patentability as described more completely hereafter.

CONCISE SUMMARY OF THE SEPARATELY-ARGUED CLAIMS

In the summaries provided below, references to the text of the specification are made parenthetically, in the following manner (Spec. Page:Lines), and references to the drawings are made in a similar manner, as follows (Fig. Number:Reference).

1. The Group I Claims

Independent claim 41 claims, inter alia, a simulation environment (Spec. 4:5-14) running on a computer system and having at least one server process (Spec. 4:5-14) (Fig. 1:120), at least one client process (Spec. 4:5-14) (Fig. 1:130), and only one control process (Spec. 4:5-14) (Fig. 1:110), where all messages between the server process and the client process are controlled by and relayed through the control process (Spec. 4:5-14), the control process sets synchronization points in the server process (Spec. 5:26-6:4), the synchronization points indicating points in time where the server process pauses until restarted by the control process (Spec. 5:26-6:4), where the server process, the client process, and the control process are all separate and distinct processes (Spec. 4:5-14) (Embodiment of Fig. 1).

2. The Group II Claims

In addition to the elements recited by independent claim 41, the representative dependent claim of the Group II claims, claim 7, claims *inter alia*, wherein the control process sets up a predetermined ordered queue of the server processes and a predetermined ordered queue of the client processes, and the messages are sent to and from the client processes and the server processes according to the predetermined ordered queues of server processes and client processes (Spec. 8:25-9:6).

3. The Group III Claims

In addition to the elements recited by independent claim 41, the representative dependent claim of the Group III claims, claim 8, claims *inter alia*, the server process evaluates an event expression to determine the occurrence of an event in the server process, and, the server process sends an event expression message to the control process upon the occurrence of the event in the server process (Spec 7:3-13), the event expression message containing a time stamp (Spec 8:3-7), the time stamp being an indication of a time at which the event occurred in the server process.

4. The Group IV Claims

In addition to the elements recited by independent claim 41, the representative dependent claim of the Group IV claims, claim 12, claims *inter alia*, the control process receives a plurality of the event expression messages from the server processes (Spec 7:3-13), the control process sorts the event expression messages received from the server processes according to the server order queue, and the control process orders each of the event expression messages within the server order queue (Spec 8:25-9:6) according to an earliest of the time stamps (Spec 8:3-7) at which the event occurred in the server process.

5. The Group V Claims

In addition to the elements recited by independent claim 41, the representative dependent claim of the Group V claims, claim 14, claims *inter alia*, the client processes each send a finish message (Spec 10:3-16), indicating the client process is finished

running, to the control process for communication to the server process associated with the client process, when the client process is finished running, the control process holds each of the finish messages from the client processes until all of the client processes associated with a server process are finished running, and the control process sends a finish message to the server process indicating the client processes are finished running.

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

(A) Whether claims 41, 7, 11, 17, and 23 are patentable under 35 USC § 103 over Kuo et al. in view of Redmond.

(B) Whether claims 8, 16, and 27 are patentable under 35 USC § 103 over Kuo et al. in view of Redmond and further in view of Baker et al.

(C) Whether claim 10 is patentable under 35 USC § 103 over Kuo et al. in view of Redmond and further in view of Trinh et al.

(D) Whether claims 12-13 are patentable under 35 USC § 103 over Kuo et al. in view of Redmond and further in view of Willmann et al.

(E) Whether claims 14-15 are patentable under 35 USC § 103 over Kuo et al. in view of Redmond and further in view of Wegrzyn.

(F) Whether claim 20 is patentable under 35 USC § 103 over Kuo et al. in view of Redmond and further in view of Schwaller et al.

(G) Whether claim 22 is patentable under 35 USC § 103 over Kuo et al. in view of Redmond and further in view of Gee.

VII. ARGUMENTS

Applicants have divided this section into groups (A) through (G), representing the seven different grounds of rejection, respectively, from section VI of this document. Applicants have further subdivided each of those seven groups into separate subsections for the claims that are argued separately within each group.

INITIAL COMMENTS ON REJECTIONS

Applicants note that the examiner has cited various combinations of references against the claims of the present application, where the references generally relate to

computer technology. Some of these references use language that is similar to the language used in the present claims. However, the examiner has not accounted for the fact that similar language is often used to refer to different logical constructs. Further, the examiner has not accounted for the fact that in the present claims these logical constructs are put together with certain specified interrelationships. The examiner appears to have found similar terms in the prior art, and having once found them, cites them against the claims as though they represent the same logical constructs and have the same interrelationships as the elements of the present claims. As described more particularly below, applicants assert that such sweeping generalizations are not accurate.

(A) WHETHER CLAIMS 41, 7, 11, 17, AND 23 ARE PATENTABLE UNDER 35 USC § 103 OVER KUO ET AL. IN VIEW OF REDMOND

1. The Group I Claims

The Group I claims are patentably distinct from the other groups because they are the broadest claims directed to at least one server process, at least one client process, and only one control process, which are all separate and distinct processes, and thus include a set of limitations that is broader than any of the other groups, as described more completely below.

Independent claim 41 claims, in relevant portion, a simulation environment running on a computer system and having at least one ***server process***, at least one ***client process***, and ***only one control process***, where all messages between the server process and the client process are ***controlled by and relayed through the control process***, the control process sets ***synchronization points*** in the server process, the synchronization points indicating points in time where the server process pauses ***until restarted by the control process***, where the server process, the client process, and the control process are ***all separate and distinct processes***.

Thus, independent claim 41 describes the following combination of elements: (1) only one control process, (2) at least one server process, (3) at least one client process, (4) where the control process, client process, and the server process are separate and distinct processes, (5) all messaging between the server process and the client process is

controlled by ***and*** relayed through the control process, and (6) the control process sets synchronization points where the server process pauses until restarted by the control process.

The combination of Kuo et al. and Redmond does not describe such a software system. Applicants first compare the primary reference (Kuo et al.) against the elements of the claim as recited above, to determine wherein the primary reference is deficient. Then the secondary reference (Redmond) is analyzed to determine whether it compensates for the deficiencies detected in the primary reference. If both of the references are deficient as to the same element or combination of elements, then the claim is patentable over the cited references.

Kuo et al. are deficient in regard to elements (4), (5), and (6) as enumerated above. ***First***, the transaction message control mechanisms of Kuo et al. reside within the server process, whereas in the claimed system the control process and the server process are separate and distinct processes. As this deficiency is acknowledged by the examiner, applicants will not further substantiate the deficiency. ***Second***, the client processes of Kuo et al. control the messaging between the server process and the client processes, whereas in the claimed system all messaging between the server process and the client process are controlled by and relayed through the control process. As this deficiency is also acknowledged by the examiner, applicants will not further substantiate the deficiency.

Third, the ***control process*** of Kuo et al. does not set synch points in the server process in the manner as presently claimed in claim 41. Instead (according to the passage referenced by the examiner), the ***output process*** 48 (which is a part of the server process 42 as depicted in Fig. 2) controls the communication from the transaction process 50 (also a part of the server process 42) to the client process 40. The output process 48 synchronizes all of the transaction processes 50. Thus, Kuo et al. describe a server process that has internal synchronization, because ***all of the control and synchronization elements are disposed within the server process***. This is very different from the present invention as claimed, where the separate and distinct ***control process sets synchronization points*** in the separate and distinct server process, and the server process stops until it is started again by the control process. The advisory action states that Kuo

clearly describes synchronizing out put from the server, but this overly simplifies the claims. It is important that the proper element provides the synchronization in order for the claim to not be patentable, and Kuo et al. do not describe the synchronization as claimed.

Thus, there are at least three patentable distinctions between the system of Kuo et al. and the system as presently claimed. Redmond must compensate for all three of these deficiencies or the claim is patentable over the combination. However, Redmond does not compensate for any of the three deficiencies.

As to the ***first deficiency*** of Kuo et al. as recited above, Redmond does teach separate and distinct message data (clients?), central controller, and databases (servers?). Thus, for the moment only, applicants will accept that Redmond arguably teaches separate and distinct client processes, server processes, and control process. However, applicants assert that Kuo et al. and Redmond are improperly combined, as described hereinafter in more detail. Therefore, the combination of Kuo et al. and Redmond does not fairly teach the separate and distinct client processes, server processes, and control process of the invention as claimed in claim 41.

In regard to the ***second deficiency*** of Kuo et al. as recited above, Redmond does not describe that all messages between the server process and the client process are controlled by ***and relayed through*** the control process – despite the simplistic drawing which might cause one to believe that the messages do pass through the central controller. Instead, Redmond describes that the central controller controls the external database ***to forward information*** matching the information request to the user (see block 114 of figure 4 and accompanying description). Thus, the central controller of Redmond ***does not relay*** all of the messages between the server process and the client process as presently claimed. Instead, the controller instructs the database to forward information directly to the user, without sending the information through the controller. In other words, the lines connecting the elements in the simplistic drawing of Redmond represent control lines, not data lines. Thus, the controller of Redmond is not performing the same function as the controller as claimed.

As to the ***third deficiency*** of Kuo et al. as recited above, Redmond is completely mute in regard to the control process setting synch points in the server processes. Thus,

Redmond does not compensate for the third deficiency of Kuo et al. Therefore, the combination of Kuo et al. and Redmond is deficient in at least the second and third deficiencies as described above, and improperly combined in regard to the first deficiency, as described in more detail hereinafter. Thus, claim 41 patentably defines over Kuo et al. in view of Redmond, and the rejection is in error.

Dependent claim 23 depends from independent claim 41, and additionally claims that ***the control process stops the server process when the server process reaches a synchronization point***. Kuo et al. describe that the return of the transaction response to the client process under control of the output process synchronizes the transaction output, but do not teach that the control process stops the server process when the server process reaches a synchronization point. Thus, even though the term "synchronization" is used in both Kuo et al. and in claim 23, the elements being synchronized and the manner in which synchronization occurs is very different between the claims and the reference. Therefore, claim 23 patentably defines over Kuo et al. in view of Redmond, and the rejection is in error.

2. The Group II Claims

The Group II claims are patentably distinct from the other groups because they are a narrower specific embodiment wherein ***ordered queues*** are implemented, and thus include a set of limitations that is of a different breadth than any of the other groups, as described more completely below.

Dependent claim 7 depends from independent claim 41, and additionally claims that the control process sets up a ***predetermined ordered queue of the server processes*** and a ***predetermined ordered queue of the client processes***, and the messages are sent to and from the client processes and the server processes according to the predetermined ordered queues of server processes and client processes.

Dependent claim 11 depends from independent claim 41, and additionally claims that the control process sets up a ***server order queue of the server processes*** and a ***client order queue of the client processes***, and the messages are sent to and from the client processes and the server processes according to the server order queue and the client order queue.

Thus, claims 7 and 11 both describe specific arrangements of ordered queues of server processes and client processes. Kuo et al. describe an input queue and an output queue, but these queues are message queues, and not queues of server processes and client processes as claimed. Once again, similar language has been found between the claims and the cited references, but the language is referring to different logical constructs.

Thus, for these reasons and the reasons given above in regard to claim 41, claims 7 and 11 patentably define over Kuo et al. in view of Redmond, and the rejections are in error.

3. The Group IV Claims

The Group IV claims are patentably distinct from the other groups because they are a narrower specific embodiment wherein ***both ordered queues and event expressions*** are implemented, and thus include a set of limitations that is of a different breadth than any of the other groups, as described more completely below.

Dependent claim 17 depends from independent claim 41, and additionally claims the control process setting up a ***client ordered queue of client processes***, a ***server ordered queue of server processes***, and a ***time ordered queue of event expression messages*** received from the server processes, the time ordered queue ordered according to an earliest in time event expression message. Neither Kuo et al. nor Redmond describe the event expression messages that indicate a user specified event as defined in the specification. Thus, for these reasons and the other reasons given above, claim 17 patentably defines over Kuo et al. in view of Redmond, and the rejection is in error.

(B) WHETHER CLAIMS 8, 16, AND 27 ARE PATENTABLE UNDER 35 USC § 103 OVER KUO ET AL. IN VIEW OF REDMOND AND FURTHER IN VIEW OF BAKER ET AL.

The rejections made under subsections (B)-(G) of this part all add a different third additional reference to the basic combination of Kuo et al. in view of Redmond. The additional "third references" are not used to compensate for the deficiencies of Kuo et al. in view of Redmond as described above, but instead are used as prior art descriptions of the additional elements that are claimed in the dependent claims. Therefore, none of the

third references describe the subject matter of the single independent claim, which includes separate and distinct processes, where all messaging between the server process and the client process is controlled by and relayed through the control process, and the control process sets synchronization points in the server process. Because of this, all of the dependent claims are allowable over the cited combinations of art without further argument. However, for the sake of completeness, and as an indication of the vast differences between the subject matter as claimed and the cited references, applicants separately argue below the dependent claims in their several groups as described elsewhere herein.

2. The Group III Claims

The Group III claims are patentably distinct from the other groups because they are a narrower specific embodiment wherein ***event expressions*** are implemented, and thus include a set of limitations that is of a different breadth than any of the other groups, as described more completely below.

Dependent claim 8 depends from independent claim 41, and additionally claims that ***the server process evaluates an event expression*** to determine the occurrence of an event in the server process, and, ***the server process sends an event expression message to the control process*** upon the occurrence of the event in the server process, the event expression message containing a time stamp, the time stamp being an indication of a time at which the event occurred in the server process.

Dependent claim 16 depends from independent claim 41, and additionally claims a plurality of client processes, each of the client processes associated with a predetermined server process, and communicating with the predetermined server process under the direction of the control process, and a plurality of server processes, each of the ***server processes evaluating an event expression*** to determine the occurrence of an event in the server process, and each of the ***server processes sending an event expression message to the control process*** upon the occurrence of the event in the server process, the event expression message containing a time stamp indicating a time at which the event occurred in the server process.

Dependent claim 27 depends from independent claim 41, and additionally claims a plurality of server processes, a plurality of client processes associated with the server processes, the plurality of server processes communicating via the control process with the client processes associated with each of the server processes, wherein each of the ***server processes evaluates an event expression*** to determine the occurrence of an event in the server process, and each of the ***server processes sends an event expression message to the control process*** upon the occurrence of the event in the server process, the event expression message contains a time stamp indicating a time at which the event occurred in the server process.

The deficiencies of Kuo et al. in view of Redmond in regard to the basic combination of elements are described above. Baker et al. do not compensate for the deficiencies of Kuo et al. in view of Redmond, in that Baker et al. also do not describe separate and distinct processes, where all messaging between the server process and the client process is controlled by and relayed through the control process, and the control process sets synchronization points in the server process.

Further, Baker et al. do not describe the evaluation of the event expressions and sending of event expression messages as claimed. Baker et al. describe that ***a client session's time stamp*** is updated each time a message transaction containing the session id for the session is received. However, this does not describe evaluation of an event expression as claimed, or the server process sending an event expression message to the control process, where ***the event expression message bears a stamp*** indicating the time at which the event occurred ***in the server process***. Once again, similar language has been found, but the ideas expressed by the language are completely different between the claims and the cited references.

Thus, claims 8, 16, and 27 patentably define over Kuo et al. in view of Redmond and further in view of Baker et al., and the rejections are in error.

(C) WHETHER CLAIM 10 IS PATENTABLE UNDER 35 USC § 103 OVER KUO ET AL. IN VIEW OF REDMOND AND FURTHER IN VIEW OF TRINH ET AL.

1. The Group I Claims

The Group I claims are patentably distinct from the other groups because they are the broadest claims directed to at least one server process, at least one client process, and only one control process, which are all separate and distinct processes, and thus include a set of limitations that is broader than any of the other groups, as described more completely below.

Dependent claim 10 depends from independent claim 41, and additionally claims that the control process maintains a time stamp for each server process, the time stamp being an indication of an elapsed time from a start of the control process, ***where the elapsed time is proportional to a time elapsed in the control process between the synchronization points***.

Trinh et al. do not describe an elapsed time that is proportional to a time elapsed in the control process between the synchronization points. Trinh et al. describe that when the presentation time and the server elapsed time differs by an unacceptably large amount, an adjustment is made to the audio data stream to re-synchronize the presentation time with the elapsed time. Thus, the time stamps are different between the claim and the cited reference. In the claim, the time is an elapsed time in the control process between synchronization points. In Trinh et al., the time difference is used to re-synchronize the presentation time with the elapsed time. Once again, similar words, but dissimilar concepts.

Thus, for these reasons and also for the reasons given above in regard to claim 41, claim 10 patentably defines over Kuo et al. in view of Redmond and further in view of Trinh et al., and the rejection is in error.

(D) WHETHER CLAIMS 12-13 ARE PATENTABLE UNDER 35 USC § 103 OVER KUO ET AL. IN VIEW OF REDMOND AND FURTHER IN VIEW OF WILLMANN ET AL.

1. The Group IV Claims

The Group IV claims are patentably distinct from the other groups because they are a narrower specific embodiment wherein ***both ordered queues and event expressions*** are implemented, and thus include a set of limitations that is of a different breadth than any of the other groups, as described more completely below.

Dependent claim 12 depends from independent claim 41 (through claim 11), and additionally claims that the control process receives a plurality of the event expression messages from the server processes, the control process ***sorts the event expression messages received from the server processes according to the server order queue***, and the control process orders each of the event expression messages within the server order queue according to an earliest of the time stamps at which the event occurred in the server process.

Dependent claim 13 depends from independent claim 41 (through claims 11-12), and additionally claims that the control process delivers the sorted event expression messages to the client processes associated with the server processes ***according to the client order queue***.

Willmann et al. teach that each of these data packets is provided with a time stamp, which gives information on the order of arrival of the data packets, and the queues are organized as FIFO queues. However, this is not what is described in claims 12 and 13. Instead, the event expression messages as claimed are sorted according to one of either the server order queue or the client order queue, but they are not sorted according to a time stamp in a FIFO queue, as described by Willmann et al. Once again, similar words are used in both the claims and in the references, but these words do not describe logical structures that have the same relationships.

Thus, for these and other reasons, claims 12-13 patentably define over Kuo et al. in view of Redmond and further in view of Willmann et al., and the rejections are in error.

(E) WHETHER CLAIMS 14-15 ARE PATENTABLE UNDER 35 USC § 103 OVER KUO ET AL. IN VIEW OF REDMOND AND FURTHER IN VIEW OF WEGRZYN.

1. The Group V Claims

The Group V claims are patentably distinct from the other groups because they are a narrower specific embodiment wherein ***finish messages*** are implemented, and thus include a set of limitations that is of a different breadth than any of the other groups, as described more completely below.

Dependent claim 14 depends from independent claim 41, and additionally claims that ***the client processes each send a finish message***, indicating the client process is finished running, to the control process for communication to the server process associated with the client process, when the client process is finished running, the control process ***holds each of the finish messages*** from the client processes until all of the client processes associated with a server process are finished running, and ***the control process sends a finish message to the server process*** indicating the client processes are finished running.

Dependent claim 15 depends from independent claim 41, and additionally claims that the ***server processes each send a finish message***, indicating the server process is finished running, to the control process when the client processes associated with each of the server processes are finished, the control process ***holds each of the finish messages*** from the server processes until all of the server processes have sent the finish messages to the control process, and the server processes, client processes, and control process finish operations and exit.

Wegrzyn et al. describe that messages are scheduled using a temporary data structure that temporarily holds messages until the end of each scheduler processing cycle, when the messages are ready to be formatted by the frame formatter. However, this does not describe a control process that receives finish messages from the client processes and the server processes, and holds them until the control process has received a finish message from each client process or each server process. Wegrzyn et al. describe holding messages until the end of a cycle, not until a message is received from each

process. As mentioned previously, the language used by Wegrzyn et al. is somewhat similar to that used in the claims, but the ideas expressed are completely different.

Thus, for these and other reasons, claims 14-15 patentably define over Kuo et al. in view of Redmond and further in view of Wegrzyn, and the rejections are in error.

(F) WHETHER CLAIM 20 IS PATENTABLE UNDER 35 USC § 103 OVER KUO ET AL. IN VIEW OF REDMOND AND FURTHER IN VIEW OF SCHWALLER ET AL.

1. The Group I Claims

The Group I claims are patentably distinct from the other groups because they are the broadest claims directed to at least one server process, at least one client process, and only one control process, which are all separate and distinct processes, and thus include a set of limitations that is broader than any of the other groups, as described more completely below.

Dependent claim 20 depends from independent claim 41, and additionally claims that the simulation environment simulates a device selected from a group consisting of electrical devices, mechanical devices, electromechanical devices, computer networks, DSL modems, ASIC disk drive controllers, graphic processors, network interface adapters, and communications networks.

Claim 20 depends from independent claim 41. Schwaller et al. does not remedy the deficiencies of Kuo et al. in view of Redmond in regard to the limitations discussed above in reference to claim 41. Thus, for these and other reasons, claim 20 patentably defines over Kuo et al. in view of Redmond and further in view of Schwaller et al., and the rejection is in error.

(G) WHETHER CLAIM 22 IS PATENTABLE UNDER 35 USC § 103 OVER KUO ET AL. IN VIEW OF REDMOND AND FURTHER IN VIEW OF GEE.

1. The Group I Claims

The Group I claims are patentably distinct from the other groups because they are the broadest claims directed to at least one server process, at least one client process, and only one control process, which are all separate and distinct processes, and thus include a

set of limitations that is broader than any of the other groups, as described more completely below.

Dependent claim 22 depends from independent claim 41, and additionally claims that the control process includes a synchronization varying module for ***varying an elapsed time duration between the synchronization points***.

Gee states that either the start time of the response or the duration of the response may be modified to cause the response to eventually overlap the hit window. However, modifying a response start time or duration is not the same thing as varying an elapsed time duration between synchronization points. The response time and duration as described by Gee do not have anything to do with the synchronization of server processes and client processes, as presently claimed. As mentioned repeatedly above, although a similar term is used, a variable time can have many different applications, and the variable time as applied in Gee is very different from the variable time as applied in claim 22. Applicants are not claimed the idea of varying a time as an abstract concept, but rather, a specific time is varied between a unique combination of elements as claimed.

In addition, claim 20 depends from independent claim 41, and Gee does not remedy the deficiencies of Kuo et al. in view of Redmond in regard to the limitations discussed above in reference to claim 41. Thus, for these and other reasons, claim 22 patentably defines over Kuo et al. in view of Redmond and further in view of Gee, and the rejection is in error.

IMPROPER COMBINATION OF REFERENCES

The claims recite certain elements in combination. Applicants do not at this time assert the claim that any one of these elements, taken by itself, is novel and has never before been seen. Thus, applicants anticipate that it might be possible to find each and every element somewhere in the prior art. Even so, applicants assert that they have combined these possibly-known elements in a novel and nonobvious manner to produce a product that has great benefits.

As mentioned repeatedly above, the examiner has found various words in the cited references that appear to be the same as those used in the claims. However, as has

been pointed out repeatedly above, these words do not refer to the same logical constructs, and do not describe elements with the same relationships as those claimed.

In addition, the examiner must provide proper motivation for making the selection and combination of prior art elements. Applicants assert that without the proper motivation, the combination of elements as recited by the examiner is not obvious. The mere fact that various elements ***could be*** placed in combination is not a sufficient motivation for actually making the combination. An infinite number of different elements ***could be*** placed in combination, but in order to make the present combination obvious, there must be an allowable motivation to make the combination.

Similarly, to just recite a benefit of the selected combination is also not sufficient. Almost every combination has one or more benefits of some type. Thus, the fact that a given combination may have a certain benefit in common with many other different combinations does nothing to make that given combination obvious over any of the other combinations. Further, the identified benefits must be obvious from the prior art, and not just in light of the present invention.

Thus, it is respectfully submitted that the references cited do not support combining the elements as claimed in the present invention. *In re Bond*, 910 F.2d 831, 15 U.S.P.Q.2d (BNA) 1566 (Fed. Cir. 1990) states that the PTO erred in rejecting a claimed invention as an obvious combination of the teaching of prior art references when the prior art provided no teaching, suggestion, or incentive supporting the combination. *See Northern Telecom Inc. v. Datapoint Corp.*, 15 U.S.P.Q.2d 1321, 1323, *In re Geiger*, 2 U.S.P.Q.2D 1276, 1278. *SmithKline Diagnostics, Inc. v. Helena Laboratories Corp.*, 859 F.2d 878, 887, 8 U.S.P.Q.2d (BNA) 1468, 1475 (Fed. Cir.1988) states that one "cannot pick and choose among the individual elements of assorted prior art references to recreate the claimed invention."

There is nothing in the prior art to lead a person of ordinary skill to design an apparatus like that of the present invention, other than the hindsight knowledge of this invention. However, the motivation to combine references cannot come from the invention itself. *See In re Oetiker*, 24 U.S.P.Q.2D 1443, 1446. The claims of the present application appear to have been used as a frame, and individual parts of separate prior art

references were employed to recreate a facsimile of the claimed invention. *See W.L. Gore & Assoc., Inc. v. Garlock, Inc.*, 220 U.S.P.Q. 303, 312.

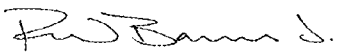
The examiner has the burden to show some teaching or suggestion in the references to support their use in the particular claimed combination. *Uniroyal Inc. v. Rudkin-Wiley Corp.*, 5 U.S.P.Q.2D at 1438-1439. In the absence of such, applicants respectfully suggest that the references are improperly combined.

CONCLUSION

In light of the many deficiencies of the rejections described above, the rejections to the claims should be reversed and the claims should be allowed.

Sincerely,

LUEDEKA, NEELY & GRAHAM, P.C.

By: 

Rick Barnes, 39,596

VIII. CLAIMS APPENDIX

1. (canceled)
2. (canceled)
3. (canceled)
4. (canceled)
5. (canceled)
6. (canceled)

7. (previously presented) The simulation environment of claim 41, wherein the control process sets up a predetermined ordered queue of the server processes and a predetermined ordered queue of the client processes, and the messages are sent to and from the client processes and the server processes according to the predetermined ordered queues of server processes and client processes.

8. (previously presented) The simulation environment of claim 41, wherein:

the server process evaluates an event expression to determine the occurrence of an event in the server process, and,

the server process sends an event expression message to the control process upon the occurrence of the event in the server process, the event expression message containing a time stamp, the time stamp being an indication of a time at which the event occurred in the server process.

9. (canceled)

10. (previously presented) The simulation environment of claim 41, wherein the control process maintains a time stamp for each server process, the time stamp being an indication of an elapsed time from a start of the control process, where the elapsed time is proportional to a time elapsed in the control process between the synchronization points.

11. (previously presented) The simulation environment of claim 41, wherein the control process sets up a server order queue of the server processes and a client order queue of the client processes, and the messages are sent to and from the

client processes and the server processes according to the server order queue and the client order queue.

12. (previously presented) The simulation environment of claim 11, wherein:

the control process receives a plurality of the event expression messages from the server processes,

the control process sorts the event expression messages received from the server processes according to the server order queue, and

the control process orders each of the event expression messages within the server order queue according to an earliest of the time stamps at which the event occurred in the server process.

13. (previously presented) The simulation environment of claim 12, wherein the control process delivers the sorted event expression messages to the client processes associated with the server processes according to the client order queue.

14. (previously presented) The simulation environment of claim 41, wherein:

the client processes each send a finish message, indicating the client process is finished running, to the control process for communication to the server process associated with the client process, when the client process is finished running,

the control process holds each of the finish messages from the client processes until all of the client processes associated with a server process are finished running, and

the control process sends a finish message to the server process indicating the client processes are finished running.

15. (previously presented) The simulation environment of claim 41, wherein:

the server processes each send a finish message, indicating the server process is finished running, to the control process when the client processes associated with each of the server processes are finished,

the control process holds each of the finish messages from the server processes until all of the server processes have sent the finish messages to the control process, and

the server processes, client processes, and control process finish operations and exit.

16. (previously presented) The simulation environment of claim 41, further comprising:

a plurality of client processes, each of the client processes associated with a predetermined server process, and communicating with the predetermined server process under the direction of the control process, and

a plurality of server processes, each of the server processes evaluating an event expression to determine the occurrence of an event in the server process, and each of the server processes sending an event expression message to the control process upon the occurrence of the event in the server process, the event expression message containing a time stamp indicating a time at which the event occurred in the server process.

17. (previously presented) The simulation environment of claim 41, further comprising the control process setting up a client ordered queue of client processes, a server ordered queue of server processes, and a time ordered queue of event expression messages received from the server processes, the time ordered queue ordered according to an earliest in time event expression message.

18. (canceled)

19. (canceled)

20. (previously presented) The simulation environment of claim 41, wherein the simulation environment simulates a device selected from a group consisting of electrical devices, mechanical devices, electromechanical devices, computer networks, DSL modems, ASIC disk drive controllers, graphic processors, network interface adapters, and communications networks.

21. (canceled)

22. (previously presented) The simulation environment of claim 41, wherein the control process includes a synchronization varying module for varying an elapsed time duration between the synchronization points.

23. (previously presented) The simulation environment of claim 41, wherein the control process stops the server process when the server process reaches a synchronization point.

24. (canceled)
25. (canceled)
26. (canceled)

27. (previously presented) The simulation environment of claim 41, wherein:

a plurality of server processes, a plurality of client processes associated with the server processes, the plurality of server processes communicating via the control process with the client processes associated with each of the server processes,

wherein each of the server processes evaluates an event expression to determine the occurrence of an event in the server process, and each of the server processes sends an event expression message to the control process upon the occurrence of the event in the server process, the event expression message contains a time stamp indicating a time at which the event occurred in the server process.

28. (canceled)
29. (canceled)

30. (withdrawn) The method of claim 42, further comprising the steps of:

determining the occurrence of predetermined events in the server processes,

maintaining with the control process a list of client processes, a list of server processes, and a list of messages associated with the predetermined events, and

communicating the associated message to the client processes upon occurrence of one of the predetermined events.

31. (withdrawn) The method of claim 30, further comprising the steps of:
ordering with the control process the messages according to an earliest time that the predetermined events occurred in the server processes, and
delivering the messages to the client processes according to the ordering.

32. (withdrawn) The method of claim 31, wherein the ordering of the messages is determined by at least one of:
(1) time order, by an earliest time that such predetermined events occurred in the server processes,
(2) server order, according to a predetermined order of server processes, and
(3) client order, according to a predetermined order of clients.

33. (withdrawn) The method of claim 32, further comprising the steps of:
sorting the messages according to the server order and the time order, and
delivering the messages from the control process to the client processes according to the client order and the time order, with earliest messages delivered first.

34. (canceled)
35. (canceled)

36. (withdrawn) The method of claim 42, further comprising the steps of:
setting a plurality of synchronization points of elapsed time in the simulation of servers and clients,
determining an occurrence of a predetermined event in the server processes,
maintaining, with the control process, a list of client processes, a list of server processes, and a list of occurrences of the predetermined events,
communicating the predetermined events to the client processes,
ordering with the control process the predetermined events according to an earliest time that the predetermined events occurred in the server processes, and
delivering messages to the client processes relating to the predetermined events according to the ordering of the predetermined events.

37. (canceled)

38. (withdrawn) The method of claim 42, further comprising the steps of:

polling each of the client processes with the control process in a predetermined manner,

temporarily storing the messages from the client processes until the client processes issue a predetermined message to simulate to the control process, and

forwarding the messages from the client processes to the server processes associated with the client processes upon the occurrence of the predetermined message to simulate.

39. (canceled)

40. (canceled)

41. (previously presented) A simulation environment running on a computer system comprising:

at least one server process capable of sending and receiving messages,

at least one client process capable of sending and receiving messages, and

only one control process for receiving the messages sent from the server process and the client process and for sending the messages sent from the server process and the client process, where all messages between the server process and the client process are controlled by and relayed through the control process,

the control process sets synchronization points in the server process, the synchronization points indicating points in time where the server process pauses until restarted by the control process,

where the server process, the client process, and the control process are all separate and distinct processes.

42. (withdrawn) A method for simulating a process in a simulation environment running on a computer system, the method comprising the sequential steps of:

a. starting only one control process within the simulation environment,

b. providing configuration information to the control process, the configuration information including a number and function of server processes to include in the simulated process, and a number and function of client processes to include in the simulated process, where the number of server processes is at least one and the number of client processes is at least one, and the number of server processes and the number of client processes are mutually independent,

c. starting the server processes included in the simulated process with the control program,

d. setting a synchronization point in the server processes with the control program,

e. sending a ready to synchronize message from each server process to the control program, indicating that the server process sending the ready to synchronize message has started,

f. continuing the method when each server process included in the simulated process has sent a ready to synchronize message to the control program,

g. starting the client processes included in the simulated process with the control program,

h. polling each client process for simulate messages with the control program, using a predetermined polling order that does not vary for a given simulation,

i. accepting the simulate messages from the client processes with the control program in response to the polling of the client processes,

j. forwarding the simulate messages from the control program to an appropriate one of the server processes,

k. when simulate messages have been received from all of the client processes included in the simulated process, processing the simulate messages with the server processes,

l. sending synchronization point reports to the control process with the server processes, indicating that each server process sending a synchronization point report has reached its synchronization point, where

35 further processing of server processes reaching their synchronization point is paused, and

m. when the control program has received synchronization point reports from all of the server processes included in the simulated process, restarting the method at step (h), until the simulated process is completed.

43. (withdrawn) The method of claim 42, further comprising, in step (m), setting a different common synchronization point in all the server processes before restarting the method.

44. (withdrawn) The method of claim 42, wherein a common synchronization point is sent to each server process.

45. (withdrawn) The method of claim 42, wherein the method further comprises between step (k) and step (l):

receiving a user specified event with at least one of the server processes,

creating an event message with a time stamp with each of the at least one server
5 processes,

sending the event message to the control program with each of the at least one server processes,

pausing further processing of each of the at least one server processes,

holding the event messages with the control process until all server processes
10 have sent one of an event message and a synchronization point report,

when the control process has received one of an event message and a synchronization point report from each server process, then sending the event messages to the client processes,

acting on the event messages with the client processes, and

15 when all event messages have been acted on by the client processes, then sending a simulate message to server processes that have not yet sent synchronization point reports, which simulate messages instruct the server processes that have not yet sent synchronization point reports to continue processing.

46. (withdrawn) The method of claim 42, wherein the simulation process is completed by:

sending finish messages to the control program with the client processes,

sending a finish message to one of the server processes when all of the client processes associated with the one of the server processes has sent finish messages,

responding to the finish message sent to the one of the server processes with the one of the server processes,

when all of the server processes have received a finish message from the control program and all of the server processes have responded to the finish message, then

exiting the server processes,

exiting the client processes, and

exiting the control process.

IX. EVIDENCE APPENDIX

There is no evidence to include in this appendix.

X. RELATED PROCEEDINGS APPENDIX

There are no related proceedings to include in this appendix.